

WHAT IS CLAIMED IS:

1 1. A method of loading data into a data store connected to a computer, the method
2 comprising the steps of:
3 identifying memory constraints;
4 identifying processing capabilities; and
5 determining a number of load and sort processes to be started in parallel based on the
6 identified memory constraints and processing capabilities.

1 2. The method of claim 1, further comprising determining a number of build
2 processes based on the number of sort processes.

3 3. The method of claim 1, wherein the number of sort processes does not exceed a
4 number of indexes to be built.

5 4. The method of claim 1, wherein the number of load processes does not exceed
6 a number of partitions to be loaded.

1 5. The method of claim 1, wherein the total number of load and sort processes does
2 not exceed processing capabilities.

3 6. The method of claim 1, wherein the memory utilized by the load and sort
4 processes does not exceed memory constraints.

5 7. The method of claim 1 wherein the number of load processes and the number of
6 sort processes each require different processing power.

1 8. The method of claim 1 wherein the number of load processes and the number of
2 sort processes each require similar processing power.

1 9. The method of claim 1 wherein the number of load processes is not equal to the
2 number of sort processes.

1 10. The method of claim 1 wherein the number of load processes is equal to the
2 number of sort processes.

1 11. The method of claim 1, wherein the number of load processes is equal to the
2 number of sort processes and which is equal to half of the processing capabilities.

1 12. The method of claim 1, wherein a number of indexes is less than half of the
2 processing capabilities, wherein a number of sort processes is equal to the number of indexes,
3 and further comprising determining that a number of load processes is the smaller of the
4 difference of the processing capabilities available for the load processes and the number of sort
5 processes, and a number of partitions.

1 13. The method of claim 1, wherein a number of partitions is less than half of the
2 processing capabilities, wherein a number of load processes is equal to the number of partitions,
3 and further comprising determining that a number of sort processes is the smaller of the
4 difference of the processing capabilities available for the sort processes and the number of load
5 processes, and a number of indexes.

1 14. The method of claim 1, wherein a number of indexes is less than the difference
2 of the total amount of available memory and the amount of memory required for a main process,
3 divided by the amount of memory required for each load and sort process, wherein a number of
4 sort processes is equal to the number of indexes, and further comprising determining that the
5 number of load processes is the smaller of the difference of a total amount of available memory
6 and the amount of memory required for the main process, and the amount of memory for each
7 sort process multiplied by the number of indexes, divided by the memory required for each load
8 process, and a number of partitions.

sub
1 b1
2 15. The method of claim 1, wherein the number of partitions is less than the
3 difference of the total amount of available memory and the amount of memory required for a
4 main process, divided by the amount of memory required for each load and sort process, wherein
5 a number of load processes is equal to the number of partitions, and further comprising
6 determining that the number of sort processes is the smaller of the difference of the total amount
7 of available memory, the amount of memory required for the main process, and the amount of
8 memory for each load process multiplied by the number of partitions, divided by the memory
9 required for each sort process, and a number of indexes.

10 16. The method of claim 1, wherein a number of load processes is equal to a number
11 of sort processes which is equal to the difference of the total amount of available memory
12 available and the amount of memory required for a main process, divided by the amount of
13 memory required for each load and sort process.
14

1 17. The method of claim 1, wherein the number of indexes is less than the difference
2 of the total amount of available memory, the amount of memory required for a main process, and
3 the amount of memory required for each load process multiplied by the processing capabilities,
4 divided by the difference of the amount of memory required for each sort process and each load
5 process, wherein a number of sort processes is equal to the number of indexes, and further
6 comprising determining that the number of load processes is the smaller of the difference of the
7 processing capabilities and the number of indexes, and a number of partitions.

1 Sub
2 B1 } 18. The method of claim 1, wherein the number of partitions is less than the
3 difference of the sum of the amount of memory required for each sort process multiplied by the
4 processing capabilities, the total amount of memory required for a main process, and the amount
5 of memory required for each load process, divided by the difference of the amount of memory
6 required for each sort process and each load process, wherein a number of load processes is equal
7 to the number of partitions, and further comprising determining that the number of sort processes
8 is the smaller of the difference of the total amount of available memory, the amount of memory
9 required for the main process, and the amount of memory for each load process multiplied by
10 the number of partitions, divided by the memory required for each sort process, and a number
11 of indexes.

12 19. The method of claim 1, wherein a number of sort processes is equal to difference
13 of the total amount of available memory, the amount of memory required for a main process, and
14 the amount of memory required for each load process, divided by the difference of the amount
15 of memory required for each sort process and each load process, and wherein the number of load
16 processes is equal to the difference of the processing capabilities and the number of sort
17 processes.

1 20. An apparatus for executing parallel load operations, comprising:
2 a computer having a data store coupled thereto, wherein the data store stores data; and
3 one or more computer programs, performed by the computer, for identifying memory
4 constraints, identifying processing capabilities, and determining a number of load and sort
5 processes to be started in parallel based on the identified memory constraints and processing
6 capabilities.

1 21. The apparatus of claim 22, further comprising determining a number of build
2 processes based on the number of sort processes.

1 22. The apparatus of claim 22, wherein the number of sort processes does not exceed
2 a number of indexes to be built.

1 23. The apparatus of claim 22, wherein the number of load processes does not exceed
2 a number of partitions to be loaded.

1 24. The apparatus of claim 22, wherein the total number of load and sort processes
2 does not exceed processing capabilities.

1 25. The apparatus of claim 22, wherein the memory utilized by the load and sort
2 processes does not exceed memory constraints.

1 26. The apparatus of claim 22, wherein the number of load processes and the number
2 of sort processes each require different processing power.

1 27. The apparatus of claim 22, wherein the number of load processes and the number
2 of sort processes each require similar processing power.

1 28. The apparatus of claim 22, wherein the number of load processes is not equal to
2 the number of sort processes.

1 29. The apparatus of claim 22, wherein the number of load processes is equal to the
2 number of sort processes.

1 30. The apparatus of claim 22, wherein the number of load processes is equal to the
2 number of sort processes which is equal to half of the processing capabilities.

1 31. The apparatus of claim 22, wherein a number of indexes is less than half of the
2 processing capabilities, wherein a number of sort processes is equal to the number of indexes,
3 and further comprising determining that a number of load processes is the smaller of the
4 difference of the processing capabilities available for the load processes and the number of sort
5 processes, and a number of partitions.

1 32. The apparatus of claim 22, wherein a number of partitions is less than half of the
2 processing capabilities, wherein a number of load processes is equal to the number of partitions,
3 and further comprising determining that a number of sort processes is the smaller of the
4 difference of the processing capabilities available for the sort processes and the number of load
5 processes, and a number of indexes.

1 33. The apparatus of claim 22, wherein a number of indexes is less than the difference
2 of the total amount of available memory and the amount of memory required for a main process,
3 divided by the amount of memory required for each load and sort process, wherein a number of
4 sort processes is equal to the number of indexes, and further comprising determining that the
5 number of load processes is the smaller of the difference of a total amount of available memory
6 and the amount of memory required for the main process, and the amount of memory for each
7 sort process multiplied by the number of indexes, divided by the memory required for each load
8 process, and on a number of partitions.

1 34. The apparatus of claim 22, wherein the number of partitions is less than the
2 difference of the total amount of available memory and the amount of memory required for a
3 main process, divided by the amount of memory required for each load and sort process is equal
4 to the number of partitions, and further comprising determining that the number of sort processes
5 is the smaller of the difference of the total amount of available memory, the amount of memory
6 required for the main process, and the amount of memory for each load process multiplied by
7 the number of partitions, divided by the memory required for each sort process, and a number
8 of indexes.

1 35. The apparatus of claim 22, wherein a number of load processes is equal to a
2 number of sort processes which is equal to the difference of the total amount of available
3 memory available and the amount of memory required for a main process, divided by the amount
4 of memory required for each load and sort process.

Cont
Sub
A2

1 36. The apparatus of claim 22, wherein the number of indexes is less than the
2 difference of the total amount of available memory, the amount of memory required for a main
3 process, and the amount of memory required for each load process multiplied by the processing
4 capabilities, divided by the difference of the amount of memory required for each sort process
5 and each load process, wherein a number of sort processes is equal to the number of indexes, and
6 further comprising determining that the number of load processes is the smaller of the difference
7 of the processing capabilities and the number of indexes and, a number of partitions.

1 37. The apparatus of claim 22, wherein the number of partitions is less than the
2 difference of the sum of the amount of memory required for each sort process multiplied by the
3 processing capabilities, the total amount of memory required for a main process, and the amount
4 of memory required for each load process, divided by the difference of the amount of memory
5 required for each sort process and each load process, wherein a number of load processes is equal
6 to the number of partitions, and further comprising determining that the number of sort processes
7 is the smaller of the difference of the total amount of available memory, the amount of memory
8 required for the main process, and the amount of memory for each load process multiplied by
9 the number of partitions, divided by the memory required for each sort process, and a number
10 of indexes.

1 38. The apparatus of claim 22, wherein a number of sort processes is equal to
2 difference of the total amount of available memory, the amount of memory required for a main
3 process, and the amount of memory required for each load process, divided by the difference of
4 the amount of memory required for each sort process and each load process, and wherein the
5 number of load processes is equal to the difference of the processing capabilities and the number
6 of sort processes.

1 39. An article of manufacture comprising a program storage medium readable by a
2 computer and embodying one or more instructions executable by the computer to perform
3 method steps for loading data into a data store connected to a computer, the method comprising
4 Sub B3 } the steps of:
5 identifying memory constraints;
6 identifying processing capabilities; and
7 determining a number of load and sort processes to be started in parallel based on the
8 identified memory constraints and processing capabilities.

1 R1.126 40.
Sub A3 42. The article of manufacture of claim 43, further comprising determining a number
of build processes based on the number of sort processes.

1 R1.126 41.
43. The article of manufacture of claim 43, wherein the number of sort processes does
2 not exceed a number of indexes to be built.

1 R1.126 42.
44. The article of manufacture of claim 43, wherein the number of load processes
2 does not exceed a number of partitions to be loaded.

1 R1.126 43.
45. The article of manufacture of claim 43, wherein the total number of load and sort
2 processes does not exceed processing capabilities.

1 R1.126 44.
46. The article of manufacture of claim 43, wherein the memory utilized by the load
2 and sort processes does not exceed memory constraints.

1 R1.126 45.
47. The article of manufacture of claim 43, wherein the number of load processes and
2 the number of sort processes each require different processing power.

1 *R1.124* ^{46.}
2 ~~48.~~ The article of manufacture of claim 43, wherein the number of load processes and
the number of sort processes each require similar processing power.

1 *R1.126* ^{47.}
2 ~~49.~~ The article of manufacture of claim 43, wherein the number of load processes is
not equal to the number of sort processes.

1 *R1.126* ^{48.}
2 ~~50.~~ The article of manufacture of claim 43, wherein the number of load processes is
equal to the number of sort processes.

1 *Cont* *Sub* *A3* *R1.126* ^{49.}
2 ~~51.~~ The article of manufacture of claim 43, wherein the number of load processes is
equal to the number of sort processes which is equal to half of the processing capabilities.

1 *R1.126* ^{50.}
2 ~~52.~~ The article of manufacture of claim 43, wherein a number of indexes is less than
half of the processing capabilities, wherein a number of sort processes is equal to the number of
indexes, and further comprising, determining that a number of load processes is the smaller of
the difference of the processing capabilities available for the load processes and the number of
sort processes, and a number of partitions.

1 *R1.126* ^{51.}
2 ~~53.~~ The article of manufacture of claim 43, wherein a number of partitions is less than
half of the processing capabilities, wherein a number of load processes is equal to the number
of partitions, and further comprising determining that a number of sort processes is the smaller
of the difference of the processing capabilities available for the sort processes and the number
of load processes, and on a number of indexes.

Copy
Sub
A3

1 R1.126

52.
54.

The article of manufacture claim 43, wherein a number of indexes is less than the difference of the total amount of available memory and the amount of memory required for a main process, divided by the amount of memory required for each load and sort process, wherein a number of sort processes is equal to the number of indexes, and further comprising determining that the number of load processes is the smaller of the difference of a total amount of available memory and the amount of memory required for the main process, and the amount of memory for each sort process multiplied by the number of indexes, divided by the memory required for each load process and, a number of partitions.

1 R1.126

53.
55.

The article of manufacture claim 43, wherein the number of partitions is less than the difference of the total amount of available memory and the amount of memory required for a main process, divided by the amount of memory required for each load and sort process, wherein a number of load processes is equal to the number of partitions, and further comprising determining that the number of sort processes is the smaller of the difference of the total amount of available memory, the amount of memory required for the main process, and the amount of memory for each load process multiplied by the number of partitions, divided by the memory required for each sort process, and a number of indexes.

R1.126

54.
56.

The article of manufacture of claim 43, wherein a number of load processes is equal to a number of sort processes which is equal to the difference of the total amount of available memory available and the amount of memory required for a main process, divided by the amount of memory required for each load and sort process.

CONF 34
SUB 34
4 A3

1 ^{R1.124} ~~55~~ 57. The article of manufacture of claim 43, wherein the number of indexes is less than
2 the difference of the total amount of available memory, the amount of memory required for a
3 main process, and the amount of memory required for each load process multiplied by the
4 processing capabilities, divided by the difference of the amount of memory required for each sort
5 process and each load process, wherein a number of sort processes is equal to the number of
6 indexes, and further comprising determining that the number of load processes is the smaller of
7 the difference of the processing capabilities and the number of indexes and, a number of
8 partitions.

1 ^{R1.126} ~~56~~ 58. The article of manufacture of claim 43, wherein the number of partitions is less
2 than the difference of the sum of the amount of memory required for each sort process multiplied
3 by the processing capabilities, the total amount of memory required for a main process, and the
4 amount of memory required for each load process, divided by the difference of the amount of
5 memory required for each sort process and each load process, wherein a number of load
6 processes is equal to the number of partitions, and further comprising determining that the
7 number of sort processes is the smaller of the difference of the total amount of available memory,
8 the amount of memory required for the main process, and the amount of memory for each load
9 process multiplied by the number of partitions, divided by the memory required for each sort
10 process, and a number of indexes.

1 ^{R1.124} ~~57~~ 59. The article of manufacture of claim 43, wherein a number of sort processes is
2 equal to difference of the total amount of available memory, the amount of memory required for
3 a main process, and the amount of memory required for each load process, divided by the
4 difference of the amount of memory required for each sort process and each load process, and
5 wherein the number of load processes is equal to the difference of the processing capabilities and
6 the number of sort processes.